Project Instructions
CMSC 35401: The Interplay of Learning and Game Theory (Autumn'22)

General Instructions

The project will count for 50% of your total grade.

The goal of this project is to apply the basic concepts and theory you learned from this course to your own research projects or to *thoroughly* survey a research field of your interest yet relevant to these course topics. A successful project could be, e.g., using the concepts or theory you learned to formulate a research question in your own research field and then try to provide an (even partial) solution. Please notice that though the course is theory-oriented, your course project does *not* have to be — i.e., applied project is equally welcome!

You are encouraged to form a team of 2-4 members to complete the project together. If needed, we are happy to help you with finding team members — please feel free to email me or the TA if so (we can also create a discord or slack discussion channel if there is a lot of demand). If you really want to do a project alone, please come to talk with us first since we would like to make sure the project is doable for one person and also has sufficient content — remember that it counts for a big portion of of your grade.

Please feel free to discuss with me or the TA (Minbiao is an expert in this general field and is conducting research on related topics) regarding, e.g., identifying a topic, references, ideas for the project, though these are not mandatory.

Notice: we completely understand that research is unpredictable. So your project does not have to show prefect results — in fact, you could even have a successful project by showing all the thing you have tried and *why they failed*. The primary goal is for you to exercise the concepts and techniques you learned in an interesting research domain. As long as we see that you have tried and spent efforts (which is usually easy to see :-0), there should not be much to worry.

Timeline and Presentation Formats

Project Proposal (5 **points**) — an at most *one-page* high-level description about who are your team members, what you would like to do and some initial thoughts/survey. This step is primarily trying to force you to form the team and brainstorm the project topic. It is due by November 3'rd (11/03) 6 pm. You will submit it to Gradescope — notably, each team member needs to submit it, even though you will be submitting the same document (since this makes it easier for grading).

Project Presentation (15 **points**) — each project will be given about $10 \sim 15$ mins to be presented during the last class of this course (i.e., 12/01 Thursday).

Project Report (30 **points**) — majority of the score is assigned to the final report in PDF format, which should be at most 7 pages with single-column, 11 pt fonts and 1 inch margin. This is due by Dec 8'th (12/08) 6 pm to Gradescope as well — notably, each team member needs to submit it, even though you will be submitting the same report.

Some Suggested Project Topics (biased by the instructor's experience)

We strongly encourage you to explore your own topics for the project. In the following I provide some suggested topics just for your information. I will update the list once more things come up to my mind.

1. Attack Online Retailers' Pricing Algorithms

   Nowadays, online retailers like Amazon, Uber, Expedia (and many others) are learning and adjusting their prices for each customers. You can do a project to: (1) identify strong evidence that they are price-discriminating different customers (not all platforms doing this, but there was records indicating that some of them do); (2) come up with ways to manipulate (theoretically or experimentally) their learning algorithms to achieve lower prices for you. This would obviously be useful not only for your research but also for your life — strongly encouraged!

2. Use multiplicative weight to solve structured games such as Stackelberg games, contract design problems, principal-agent problems and security games.

   See this paper [GHWX22] for a general framework, which includes Stackelberg games, contract design, Bayesian persuasion as special cases. Another interesting class of games is the high impactful *security game*, which is a fundamental resource allocation game with significant real-world impact. See, e.g., this paper [Xu16] for an introductory reading. Previous research mostly used linear programming or OR techniques to solve security games. You may think about using the multiplicative weight update algorithm to solve the game, and see whether it is faster than classical OR-based algorithms. This survey paper [AHK12] shows how powerful multiplicative weight is.

3. The study of convergence of no-regret learning algorithms to equilibriua in structured games

   We looked at convergence of no-regret learning algorithm in zero-sum games, and some of you figured out that it also provably converges in 2×2 matrix games. Is there any other structured settings — e.g., a game arised in your research domain or a special game classes such as Stackelberg games, principal-agent problems [GHWX22], congestion games, or dominance elimination solvable games, etc. — that no-regret learning algorithm also provably or empirically converges to equilibrium? If the algorithm does not convergence, what is the patten of the its trajectory?

   There has been extensive literature on this frontier for zero-sum games [PP16, MJS19], congestion games [CXFD22], dominance-elimination solvable games [WXY22] (which turns out to be related to rationalizable equilibria [WKBJ22]). However, many problems remain open. For example, the regret bound of [CXFD22] leaves significant room for improvement. An interesting open problem from [WXY22] is whether there exists a no-regret learning algorithm that provably converges to rationalizable equilibrium in two-player games.

4. The above project is about no-regret learning, which is a special class of reinforcement learning (RL) algorithms. You can ask the same question for other RL algorithms, e.g., Q-learning or deep Q-learning. See, e.g., this paper [LZL$^+$17].

5. You may ask a more applied research question. That is, can you design fully applied algorithm — e.g., a deep learning algorithm by carefully constructing the network architecture to suit game-theoretic applications — to solve the listed games above or beyond.

6. In the first lecture (and later lectures), we studied deceptive behaviors or *strategic attacks* to ML algorithms used in economic decision making. If you work in ML, particularly in E-commerce domains, you may think about tackling situations with similar strategic manipulations.

7. Algorithms for robust Bayesian persuasion, and the potentially over conservativeness of the robust solution.

   A nice recent paper at *Econometrica* [DP20] developed a refined solution concept for the Bayesian persuasion problem, termed "robust Bayesian persuasion". Specifically, the paper enriches the standard Bayesian persuasion model by assuming the existence of an adversarial who may reveal any possible additional information simply to minimizes the sender's utility. The paper proposes to maximizes the worst sender's utility in this case, and pick the best one if there are multiple optimal solutions. This raises natural computational question of how to efficiently find such a robust solution. Moreover, how much sender utility can the receiver guarantee in such robust persuasion — is the robust solution to conservative in the sense that the sender's utility will be too small due to the existence of such a strong adversary?

8. Incentive design in Bitcoin mining. One recent hot trend is to understand gaming phenomenon in bitcoin mining and how to design effective mechanisms to incentive the right mining behaviors (e.g., transaction fee design). See, e.g., [CS21, Rou21]. You can do a survey or study new questions on this frontier.

References

[AHK12] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of computing*, 8(1):121–164, 2012.

[CS21] Hao Chung and Elaine Shi. Foundations of transaction fee mechanism design. *arXiv preprint arXiv:2111.03151*, 2021.

[CXFD22] Qiwen Cui, Zhihan Xiong, Maryam Fazel, and Simon S Du. Learning in congestion games with bandit feedback. *arXiv preprint arXiv:2206.01880*, 2022.

[DP20] Piotr Dworczak and Alessandro Pavan. Preparing for the worst but hoping for the best: Robust (bayesian) persuasion. 2020.

[GHWX22] Jiarui Gan, Minbiao Han, Jibang Wu, and Haifeng Xu. Optimal coordination in generalized principal-agent problems: A revisit and extensions. *arXiv preprint arXiv:2209.01146*, 2022.

[LZL+17] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 464–473, 2017.

[MJS19] Eric V Mazumdar, Michael I Jordan, and S Shankar Sastry. On finding local nash equilibria (and only local nash equilibria) in zero-sum games. *arXiv preprint arXiv:1901.00838*, 2019.

[PP16] Christos Papadimitriou and Georgios Piliouras. From nash equilibria to chain recurrent sets: Solution concepts and topology. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pages 227–235, 2016.

[Rou21] Tim Roughgarden. Transaction fee mechanism design. *ACM SIGecom Exchanges*, 19(1):52–55, 2021.

[WKBJ22] Yuanhao Wang, Dingwen Kong, Yu Bai, and Chi Jin. Learning rationalizable equilibria in multiplayer games. 2022.

[WXY22] Jibang Wu, Haifeng Xu, and Fan Yao. Multi-agent learning for iterative dominance elimination: Formal barriers and new algorithms. In Po-Ling Loh and Maxim Raginsky, editors, *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pages 543–543. PMLR, 02–05 Jul 2022.

[Xu16] Haifeng Xu. The mysteries of security games: Equilibrium computation becomes combinatorial algorithm design. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 497–514, 2016.